HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P. O. Box 272400
Fort Collins, Colorado 80527-2400

#18

PATENT APPLICATION

AF/2100

ATTORNEY DOCKET NO. 200304314-1

#8

ORIGINAL

03-23-04

IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Darrell R. COMMANDER

Confirmation No.: 6193

Application No.: 09/301,885

Examiner: Syed J. Ali

Filing Date: 04/29/1999

Group Art Unit: 2127

Title: A DISTRIBUTED COMPUTER NETWORK WHICH SPAWNS INTER-NODE PARALLEL PROCESSES BASED ON RESOURCE AVAILABILITY

Mail Stop Appeal Brief-Patents
Commissioner For Patents
PO Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL BRIEF

Sir:

Transmitted herewith in **triplicate** is the Appeal Brief in this application with respect to the Notice of Appeal filed on 01/16/2004.

The fee for filing this Appeal Brief is (37 CFR 1.17(c)) $330.00.

(complete (a) or (b) as applicable)

The proceedings herein are for a patent application and the provisions of 37 CFR 1.136(a) apply.

( ) (a) Applicant petitions for an extension of time under 37 CFR 1.136 (fees: 37 CFR 1.17(a)-(d) for the total number of months checked below:

( )	one month	$110.00
( )	two months	$420.00
( )	three months	$950.00
( )	four months	$1480.00

( ) The extension fee has already been filled in this application.

(X) (b) Applicant believes that no extension of time is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition and fee for extension of time.

Please charge to Deposit Account **08-2025** the sum of **$330.00**. At any time during the pendency of this application, please charge any fees required or credit any over payment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.16 through 1.21 inclusive, and any other sections in Title 37 of the Code of Federal Regulations that may regulate fees. A duplicate copy of this sheet is enclosed.

(X) I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, Alexandria, VA 22313-1450. Date of Deposit: 03/12/2004

OR

( ) I hereby certify that this paper is being transmitted to the Patent and Trademark Office facsimile number________ on ________

Number of pages:

Typed Name: **Christina L. Paz**

Signature: Christina Paz

Respectfully submitted,

Darrell R. COMMANDER

By

Jonathan M. Harris
Attorney/Agent for Applicant(s)
Reg. No. **44,144**

Date: **03/12/2004**

Telephone No.: **(713) 238-8045**

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Darrell R. COMMANDER	§	Confirmation No.:	6193
Serial No.:	09/301,885	§	Group Art Unit:	2127
Filed:	04/29/1999	§	Examiner:	Syed J. Ali
For:	A Distributed Computer Network Which Spawns Inter-Node Parallel Processes Based On Resource Availability	§	Docket No.:	200304314-1

APPEAL BRIEF

Mail Stop Appeal Brief – Patents
Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Date: March 12, 2004

Sir:

Appellant hereby submits this Appeal Brief in connection with the above-identified application. A Notice of Appeal was filed on January 16, 2004.

I. REAL PARTY IN INTEREST

The real party in interest is the Hewlett-Packard Company through its merger with Compaq Computer Corporation which owned Compaq Information Technologies Group, L.P. (CITG).

II. RELATED APPEALS AND INTERFERENCES

Appellant is unaware of any related appeals or interferences.

III. STATUS OF THE CLAIMS

Originally filed claims:	1-30.
Claim cancellations:	25-26.
Added claim:	None.
Presently pending claims:	1-24, 27-30.
Presently appealed claims:	1-24, 27-30.

IV. STATUS OF THE AMENDMENTS

Applicants did not file any amendments after the final rejection.

V. SUMMARY OF THE INVENTION

A system comprising multiple processors is generally capable of concurrently executing multiple processes. As such, each processor may be assigned a process to execute concurrently with some or all other processors. One of the issues to be addressed when processing data in a parallel processing network is how to assign processes to the various processors in the network. Applicant's Disclosure, pages 1-4.

Various embodiments of the invention comprise logic that determines whether the system has a sufficient number of processors available to spawn a desired number of processors. If not, the logic spawns a reduced number of processes. Claim 1, for example, is directed to a parallel processing network in which one or more processes can be spawned. The claimed system comprises a plurality of computers coupled together by a communications link and process spawning logic included in one of said plurality of computers. The spawning logic automatically spawns processes in response to user specified criteria and determines whether sufficient processors are available to spawn the processes. If a sufficient number of processors are not available, the spawning logic spawns a reduced number of processes based on the number of available processors.

VI. ISSUE(S)

1. Whether claims 1-3 and 5 are obvious over U.S. Patent No. 6,314,114 (*Coyle et al.*) in view of U.S. Patent No. 6,560,634 (*Broadhurst*) and U.S. Patent No. 6,182,109 (*Sharma et al.*);

2. Whether claims 4, 6-9, and 12-14 are obvious over *Coyle* in view of *Broadhurst*, *Sharma*, and U.S. Patent No. 6,173,*246 (Billups, III*);

3. Whether claims 10-11 and 16-19 are obvious over *Coyle* in view of *Broadhurst*, *Sharma*, *Billups*, and U.S. Patent No. 5,910,951 (*Pearce et al.*);

4. Whether claims 15 and 20-23 are obvious over *Coyle* in view of *Broadhurst*, *Sharma*, and *Pearce*;

5. Whether claims 24-26 are obvious over *Broadhurst* in view of *Sharma* and *Pearce*;

6. Whether claims 27-28 are obvious over *Broadhurst* in view of *Sharma, Pearce*, and *Billups;* and

7. Whether claims 29-30 are obvious over *Broadhurst* in view of U.S. Patent No. 6,108,309 *(Cohoe)*.

VII. GROUPING OF CLAIMS

- Claims 1-4, 6-18, 20-24, and 27-30 stand together (representative claim is claim 1); and
- Claims 5 and 19 stand together (representative claim is claim 5).

The groupings above are for purposes of this appeal only. The groupings should not be construed to mean the patentability of any of the claims may be determined (*e.g.*, in later actions before a court) based on the groupings. Rather, the presumption of 35 U.S.C. § 282 shall apply to each claim individually.

VIII. ARGUMENT

The Examiner rejected claims 1 and 5 over the combination of Coyle, Broadhurst and Sharma. Each reference is summarized below.

A. The Coyle Reference

The Coyle reference is directed to distributed resource management in which process pool managers 204, 208, 212 (Figure 2) spawn server processes from process pools when needed by a client process that requires work to be performed. Col. 3, lines 1-50. "As additional processing resources are required, process pool managers 204, 208, 212 can spawn additional server processes. Alternatively, as fewer processing resources are required, process pool managers 204, 208 and 212 can reduce the number of active server processes." Col. 3, lines 54-58. In Coyle, the reduction in server processes is keyed to demand for processes and not based on the availability of resources in the system.

B. The Broadhurst Reference

The Broadhurst reference is directed to a method of determining unavailability of an Internet domain name. See *e.g.*, the Title of Broadhurst. Broadhurst teaches searching existing domain name records in various domains and then displaying the results in a formatted manner, thereby eliminating the need for a user to perform individual searches. See Abstract. The Examiner was particularly drawn to the bottom portion of column 5 of Broadhurst. That portion teaches calculating a number of sub-processes to spawn to perform the search based on the number of domains to query submitted by the client computer. The calculation of the number of sub-processes is intended to limit each search sub-process to having to search no more than 11 domains. Col. 5, lines 48-54. Further, Broadhurst teaches that, "depending on the load of query search 104, search engine 226 may change the number of search sub-processes to spawn." Col. 5, lines 62-64.

C. The Sharma Reference

Sharma is directed to a server system that dynamically manages a pool of execution units. As client requests for service are received by the server system, the server determines whether there are a sufficient number of execution units to satisfy the request. If there are not a sufficient number of execution units, the client request is simply rejected and not performed. Col. 2, lines 38-52.

D. The Examiner Erred in Rejecting Claim 1

The Examiner rejected claim 1 as being obvious over the combination of Coyle, Broadhurst, and Sharma. Claim 1 is directed to a parallel processing network that includes, among other features, process spawning logic that "determines whether sufficient processors are available to spawn the processes and, if not, spawns a reduced number of processes based on the number of available processors."

The Examiner correctly observed that Coyle and Sharma do not teach the reduction of processes based on there being an insufficient number of available processors. The Examiner, however, contended that Broadhurst teaches the reduction of processes based on the number of available resources. For the following reasons, Applicant respectfully submits that the Examiner erred in this analysis.

First, Broadhurst teaches that "depending on the load of query server 104, search engine 226 may change the number of search sub-processes to spawn"

(Col. 5, lines 62-64). The load of query server 104 does not represent or even suggest the number of available processors, as is required by claim 1. The load of a server is different than the number of available processors. Accordingly, Broadhurst does not teach changing the number of search sub-processes to spawn based on the number of available processors.

Second, the query server in the Broadhurst reference is a single processor system (Col. 4, line 1). As is required by claim 1, the reduction in processes is based on the number of available processors. Thus, as long as at least one processor is available, the process spawning logic will spawn a reduced number of processes depending on the number of processors that are available. Broadhurst does not teach or even imply the use of more than one processor. Thus, Broadhurst cannot, and accordingly does not, teach reducing the number of processes based on the number of available processors.

For any or all of the foregoing reasons, Applicant respectfully submits that the Examiner erred in rejecting claim 1. For at least these same reasons, the Examiner erred in rejecting all claims that depend on or from claim 1, as well as all other claims in the same group as claim 1 (see claim grouping above).

E. The Examiner Erred in Rejecting Claim 5

The Examiner rejected claim 5 as being obvious over the combination of Coyle, Broadhurst, and Sharma. Claim 5 depends from claim 1 and is allowable for at least the reasons provided above for claim 1. Further, claim 5 also requires that the "user specified criteria also includes a maximum number of CPUs to be used per machine to execute processes."

The Examiner conceded that Coyle and Broadhurst do not teach the use of a maximum number of CPUs to be used per machine. However, the Examiner contended that Sharma teaches the use of the maximum number of CPUs to be used per machine. However, Sharma teaches that the "maximum number of execution units is an upper bound to support a peak client load without overloading the server system." (Col. 2, lines 44-46) Sharma does not teach, suggest, or even imply that the user parameters include the maximum number of

CPUs, as is required by claim 5. For this additional reason, Applicant respectfully contends that the Examiner erred in rejecting claim 5.

IX. CONCLUSION

Applicant respectfully requests that the rejections in the Final Office Action be reversed and the case allowed. If any fees or time extensions are inadvertently omitted or if any fees have been overpaid, please appropriately charge or credit those fees to Hewlett-Packard Company Deposit Account Number 08-2025 and enter any time extension(s) necessary to prevent this case from being abandoned.

Respectfully submitted,

Jonathan M. Harris
PTO Reg. No. 44,144
CONLEY ROSE, P.C.
(713) 238-8000 (Phone)
ATTORNEY FOR APPLICANTS

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
Legal Dept., M/S 35
P.O. Box 272400
Fort Collins, CO 80527-2400

APPENDIX TO APPEAL BRIEF

CURRENT CLAIMS

1. (Previously presented) A parallel processing network in which one or more processes can be spawned, comprising:

a plurality of computers coupled together by a communications link; and

process spawning logic included in one of said plurality of computers that automatically spawns processes in response to user specified criteria and that determines whether sufficient processors are available to spawn the processes and, if not, spawns a reduced number of processes based on the number of available processors.

2. (Original) The parallel processing network of claim 1 wherein the communications link includes a switch.

3. (Original) The parallel processing network of claim 1 wherein the user specified criteria includes a number of processes the spawning logic should spawn.

4. (Previously presented) The parallel processing network of claim 3 wherein the user specified criteria also includes a model parameter.

5. (Previously presented) The parallel processing network of claim 3 wherein the user specified criteria also includes a maximum number of CPUs to be used per machine to execute processes.

6. (Previously presented) The parallel processing network of claim 4 wherein each of the plurality of computers includes a CPU and the model parameter refers to the type of CPU.

7. (Original) The parallel processing network of claim 3 wherein the user specified criteria includes a resource parameter.

8. (Original) The parallel processing network of claim 7 wherein each of said plurality of computers includes a network interface and the resource parameter refers a type of network interface.

9. (Original) The parallel processing network of claim 1 wherein said process spawning logic compares the user specified criteria to network features.

10. (Original) The parallel processing network of claim 9 wherein the network features are maintained in a process scheduler included in one of said plurality of computers.

11. (Original) The parallel processing network of claim 9 wherein the network features include an identification of which of said plurality of computers is operational and which are nonoperational and the spawning logic.

12. (Original) The parallel processing network of claim 9 wherein each of said plurality of computers includes a CPU and the network features include the model of CPU.

13. (Original) The parallel processing network of claim 9 wherein each of said plurality of computers includes a network interface resource and the network features include the type of network interface resource.

14. (Original) The parallel processing network of claim 9 wherein the user specified criteria includes a number of processes to be spawned and, if said spawning logic determines there are insufficient network features to spawn processes in accordance with the user specified criteria, the spawning logic spawns fewer processes than the user specified number of processes.

15. (Previously presented) A parallel processing network, comprising:

a plurality of processors coupled together by a communications link;

a process scheduler accessible by at least one of said processors, said process scheduler maintains a list of network features;

spawning logic coupled to said process scheduler, said spawning logic receives a set of parameters from a user that determine how processes are to be spawned by the root machine, the set of parameters including a user desired number of processes to be spawned, said spawning logic determines whether a sufficient number of processors are available to permit the user desired number of processes to be spawned in accordance with the user specified parameters and, if not, spawns a reduced number of processes based on the number of available processors.

16. (Original) The parallel processing network of claim 15 wherein the user parameters include a particular model of processor to which the processes are to be spawned.

17. (Original) The parallel processing network of claim 16 wherein the user parameters include a particular type of a network resource.

18. (Original) The parallel processing network of claim 17 wherein the spawning logic determines whether sufficient network features are available to permit the user desired number of processes to be spawned by accessing the process scheduler to read the list of network features.

19. (Original) The parallel processing network of claim 17 wherein the user parameters include a maximum number of CPUs to use per machine for spawning processes.

20. (Previously presented) A computer readable storage medium for storing an executable set of software instructions which, when inserted into a host computer system, is capable of controlling the operation of the host computer, said

software instructions being operable to automatically spawn parallel processes in a parallel processing network, comprising:

a means for receiving user specified criteria;

a means for reading a process scheduler to access a list of features associated with the parallel processing network;

a means for comparing the list of network features to the user specified criteria; and

a means for spawning a number of processes that is reduced based on the number of available CPUs.

21. (Original) The computer readable storage medium of claim 20 wherein the user specified criteria includes a user desired number of processes to be spawned and said means for spawning processes includes a means for spawning the user desired number of processes if said means for comparing determines that the parallel processing network has sufficient features in accordance with the user specified criteria.

22. (Original) The computer readable storage medium of claim 21 wherein said means for spawning processes includes spawning fewer than the user desired number of processes if said means for comparing determines that the parallel processing network has insufficient features in accordance with the user specified criteria.

23. (Original) The computer readable storage medium of claim 21 wherein said means for spawning processes includes spawning fewer than the user desired number of processes if said means for comparing determines that the parallel processing network has insufficient CPUs to spawn the user desired number of processes.

24. (Previously presented) A method of creating processes in a multi-processor network, comprising:

receiving criteria that determine how the processes are to be created, the criteria including a desired number of processes to be created;

comparing the criteria to a database of network features to determine if there are a sufficient number of processors to accommodate the desired number of processes; and

creating processes if a sufficient number of processors are available and, if a sufficient number of processors are not available, creating a reduced number of processes.

25. (Canceled).

26. (Canceled).

27. (Previously presented) The method of claim 24 wherein receiving criteria includes receiving criteria that also include a model of processor and a resource type for running processes.

28. (Original) The method of claim 27 wherein the resource type includes a network interface resource type.

29. (Previously presented) A method for spawning processes in a multiprocessor network, comprising:

specifying whether processes are to be spawned automatically to match a set of criteria or spawned in accordance with a process group file;

spawning processes to match the criteria if automatic spawning is specified;

spawning processes in accordance with the process group file if so specified; and

spawning a reduced number of processes automatically or in accordance with the process group file based on a number of available processors.

30. (Previously presented) The method of claim 29 further including determining whether the multiprocessor network matches the set of criteria if automatic spawning is specified.